What is claimed is:

1. A context-selection mechanism for selecting a best context from a pool of contexts for processing a data packet comprising:

an interface for communicating with a multi-streaming processor;

circuitry for computing input data into a result value according to logic rule and for selecting a context based on the computed value; and

a loading mechanism for preloading the packet information into the selected context for subsequent processing;

characterized in that the computation of the input data functions to enable identification and selection of a best context for processing a data packet according to the logic rule at the instant time such that a multitude of subsequent context selections over a period of time acts to balance load pressure on functional units housed within the multi-streaming processor and required for packet processing.

2. The context-selection mechanism of claim 1 integrated to a data packet router operating in a data-packet-network.

3. The context-selection mechanism of claim 2 wherein the data-packet-network is the Internet network.

4. The context-selection mechanism of claim 1 wherein the pool of contexts is divided into separate clusters in the processing unit, each cluster containing some of the functional units used in packet processing.

5. The context-selection mechanism of claim 1 wherein the input data into the computation circuitry includes availability information of individual ones of the pool of contexts at the time of computation.

6. The context-selection mechanism of claim 5 wherein the input data into the computation circuitry further includes real time information of any processing streams stalled in un-available ones of the pool of contexts and the reason for the stall.

7. The context-selection mechanism of claim 5 wherein the input data into the computation circuitry further includes statistical data about previous processing time periods required to process similar data packets.

8. The context-selection mechanism of claim 5 wherein the input data into the computation circuitry further includes statistical data about the distribution of instruction types associated with individual ones of previously processed and similar data packets.

9. The context-selection mechanism of claim 1 wherein the input data is sourced from the multi-streaming processor.

10. The context-selection mechanism of claim 1 wherein the input data is sourced from a third party.

11. The context-selection mechanism of claim 4 wherein the clusters are numbered and the functional units are distributed symmetrically therein.

12. The context-selection mechanism of claim 4 wherein the clusters are numbered and the functional units are distributed asymmetrically therein.

13. A system for load balancing pressure on functional units within a multi-streaming processor during the processing of multiple data packets comprising:

a context-selection mechanism having a communication interface, circuitry for computing input data according to a logic rule and a mechanism for preloading packet information into available ones of a pool of contexts;

a multi-streaming processor responsible for processing the data packets, the processor hosting the functional units and the context pool; and

a set of instructions comprising the logic rule governing context selection, wherein pressure upon the functional units within the processor core is balanced by selecting individual contexts according to the computed value following the set of instructions.

14. The system of claim 1 integrated to a data packet router operating in a data-packet-network.

15. The system of claim 2 wherein the data-packet-network is the Internet network.

16. The system of claim 1 wherein the pool of contexts is divided into separate clusters in the processing unit, each cluster containing some of the functional units used in packet processing.

17. The system of claim 1 wherein the input data into the computation circuitry includes availability information of individual ones of the pool of contexts at the time of computation.

18. The system of claim 13 wherein the input data into the computation circuitry further includes real time information of any processing streams stalled in un-available ones of the pool of contexts and the reason for the stall.

19. The system of claim 13 wherein the input data into the computation circuitry further includes statistical data about previous processing time periods required to process similar data packets.

20. The system of claim 13 wherein the input data into the computation circuitry further includes statistical data about the distribution of instruction types associated with individual ones of previously processed and similar data packets.

21. The system of claim 13 wherein the input data is sourced from the multi-streaming processor and provided in a software table.

22. The system of claim 13 wherein the input data is sourced from a third party.

23. The system of claim 16 wherein the clusters are numbered and the functional units are distributed symmetrically therein.

24. The system of claim 16 wherein the clusters are numbered and the functional units are distributed asymmetrically therein.

25. The system of claim 13 wherein the set of instructions comprising the logic rule is programmable.

26. A method for load balancing pressure on functional units contained within a multi-streaming processor core during processing of multiple data packets comprising steps of:

(a) arranging the functional units into more than one separate cluster on the core of the processor, each cluster containing an equal number of contexts that may write to the functional units within the hosting cluster;

(b) receiving a data packet for processing;

(c) receiving as input for computation, data about the instant availability status of individual contexts within each cluster;

(d) receiving as input for computation, data about stream status of streams occupying any contexts within each cluster; and

(e) computing the data received as input to produce a value, the value identifying and initiating selection of a best context for processing the data packet and balancing the load of the functional units within each cluster; and

(f) repeating steps (b) through (e) for each of the multiple data packets for processing.

27. The method of claim 26 practiced in conjunction with a data packet router operating in a data-packet-network.

28. The method of claim 27 wherein the data-packet-network is the Internet network.

29. The method of claim 26 wherein in step (a) the functional units are provided within each cluster in a symmetrical fashion.

30. The method of claim 26 wherein in step (a) the functional units are provided within each cluster in an asymmetrical fashion.

31. The method of claim 26 wherein in step (b) the packet is received at a data port of a data router and requires automatic activation.

32. The method of claim 26 wherein in step (b) the packet is held by the processor and requires a context for processing.

33. The method of claim 26 wherein in step (c) availability status comprises an indication of which one of two components own each context.

34. The method of claim 33 wherein in step (c) one of the components is the processor and other component is a packet management unit.

35. The method of claim 26 wherein in step (d) the data about stream status includes whether or not streams are stalled within any of the contexts and the reason for each instance of a stalled stream.

36. The method of claim 26 wherein in step (d) the data about stream status includes time parameters of how long each stream will take to process data packets associated with their contexts.

37. The method of claim 26 wherein in step (d) the data about stream status includes distribution parameters of instruction types that each stream has executed to process its data packet.

38. The method of claim 26 wherein in steps (c) through (d) are practice according to the rule of logic

39. The method of claim 39 wherein the rule of logic is programmable.

5